Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) Method for authenticating clients in a client-server environment, wherein said client-server environment uses a communication protocol that allows extensions of ~~the header request~~ a request header without violating said communication protocol, wherein said method comprises the steps of:

generating a ~~header~~ request header at a client computer,

inserting client authentication information into said ~~header~~ request header at a client computer by a client browser, without violating HTTP protocol, resulting in an extended ~~header~~ request header independently of ~~the~~ an authentication process used by ~~said~~ a server and without said server requesting authentication information,

sending said extended ~~header~~ request header to ~~a~~ said server,

and receiving information from said server if authentication has been successful.

2. (canceled)

3. (currently amended) Method according to claim 1, wherein said authentication information is included in ~~the first header request~~ a first request header for establishing a session with said server.

4. (currently amended) Method according to claim 1, wherein said authentication information comprises ~~the~~ a client certificate containing client's name and client public key, and a digital signature which has been generated over a hash value of the ~~header~~ request header including client certificate using Client private key.

5. (currently amended) Method according to claim 1, wherein said authentication information is automatically inserted into said ~~header~~ request header by the ~~client's~~ client browser.

6. (previously presented) Method according to claim 5, wherein said client browser receives said authentication information from a smart card via a smart card reader.

7. (currently amended) Method according to claim 1, wherein said authentication information is automatically inserted into said ~~header~~ request header by a client signature component which receives said authentication information from a smart card via a smart card reader.

8. (currently amended) Method for authenticating clients in a client-server environment, wherein said client-server environment uses a communication protocol that allows extensions of ~~the header request~~ a request header without violating said communication protocol, wherein a system establishes communication between ~~said~~ a client and ~~said~~ a server, wherein said method comprises the steps of:

receiving a ~~header~~ request header from said client,

inserting authentication information into said ~~header~~ request header at a client computer by a client browser, without violating HTTP protocol, resulting in an extended ~~header~~ request header independently of ~~the~~ an authentication process used by said server and without said server requesting authentication information,

sending said extended ~~header~~ request header to ~~a~~ said server, and

receiving information from said server, if ~~the~~ authentication has been successful.

9. (previously presented) Method according to claim 8, wherein said system can be a proxy server, a gateway, or a tunnel.

10. (currently amended) Method according to claim 8, wherein ~~said communication protocol is the HTTP-protocol, and~~ said authentication information is automatically

inserted into said HTTP-request header by ~~said~~ an insertion component which receives said authentication information from a signature component.

11. (currently amended) Method according to claim 8, wherein said authentication information comprises ~~the~~ a client certificate containing ~~client's~~ a name and ~~client's~~ a public key of the client, and a digital signature which has been generated over the whole ~~header~~ request header including the client certificate using ~~Client's~~ a private key of the client.

12. (currently amended) Method for authenticating clients in a client-server environment, wherein said client-server environment uses a communication protocol that allows extensions of ~~the header request~~ a request header without violating said communication protocol, wherein at ~~said~~ a server side said method comprises the steps of:

receiving a client ~~header~~ request header generated at a client computer, the request header containing authentication information inserted into the request header by the client computer at a client browser, without violating HTTP protocol,

validating said authentication information contained in said ~~header~~ request header by ~~said~~ a server authentication component, and

providing information to said client, if ~~the~~ an authentication has been successful.

13. (currently amended) Method according to claim 12, wherein said authentication information comprises ~~the~~ a client certificate containing ~~client's~~ a name and ~~client's~~ a public key of the client, and a digital signature which has been generated over the whole ~~header~~ request header content using ~~Client's~~ a private key of the client.

14. (currently amended) Method according to claim 12, wherein ~~said communication protocol is the HTTP protocol, and~~ said server authentication component performs the steps of:

accessing ~~said~~ a public key contained in ~~the~~ a client certificate, decrypting ~~said~~ a digital signature contained in the HTTP-request header with said public key using a hash algorithm resulting in a hash value,

applying the same hash algorithm as used by said client to said HTTP-request header, and

considering authentication as successful, if both hash values match.

15. (currently amended) Server System for authenticating clients in a ~~client service~~ client-server environment, wherein said client-server environment uses a communication protocol that allows extensions of ~~the header request~~ a request header without violating said communication protocol, wherein ~~said~~ a client provides authentication information in the ~~header~~ request header to ~~said~~ a server system, wherein said server system comprising:

a server machine configured to receive the request header,

an authentication component to operate on the server machine and with ~~the~~ functionality to read said authentication information contained in the ~~incoming client header~~ request header, and to validate said authentication information without having requested said authentication information from said client,

wherein the request header is generated by the client and the authentication information is inserted into the request header at the client by a client browser, without violating HTTP protocol.

16. (currently amended) Client System to be authenticated by a server system in client-server environment, wherein said client-server environment uses a communication protocol that allows extensions of ~~the header request~~ a request header without violating said communication protocol, wherein said client system comprises:

a browser operating on a client computer, and

a component operating on the browser for inserting client authentication information into said ~~header~~ request header independently of ~~the~~ an authentication process used by said server and without server requesting authentication information, without violating HTTP protocol.

17. (currently amended) Client System according to claim 16, wherein said authentication information comprises ~~the~~ a client certificate containing ~~client's~~ a name and ~~client's~~ a public key of the client, and a digital signature which has been generated over ~~the~~ a hash value of the ~~header~~ request header content using ~~Client's~~ a private key of the client.

18. (currently amended) Client System according to claim 16, further comprising

a smart card reader, and

a smart card with a security module containing ~~client's~~ a private key of the client and a client certificate containing a client name and a private key, wherein said smart card provides said client certificate together with a digital signature to said inserting component, wherein said digital signature is the result of an encryption of a hash value of said ~~header~~ request header containing said client certificate ~~information~~ by means of said private key.

19. (canceled)

20. (currently amended) Computer program product comprising a storage media for storing program instructions, said program instructions, when executed on a computer, causing the computer to perform a method for authenticating clients in a client-server environment, wherein the client-server environment uses a communications protocol that allows extensions of a ~~header~~ request header, said method comprising the steps of:

generating a ~~header~~ request header at a client computer,

inserting client authentication information into said ~~header~~ request header at a client computer by a client browser, without violating HTTP protocol, resulting in an extended ~~header~~ request header independently of ~~the~~ an authentication process used by ~~said~~ a server and without the server requesting authentication information,

sending said extended ~~header~~ request header to ~~a~~ the server,

and receiving information from said server if an authentication has been successful.